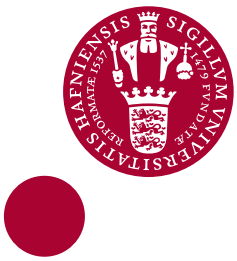

It’s a Wrap:
Mapping On-Skin Input to Off-Skin Displays

Bergstrom-Lehtovirta, Joanna; Hornbæk, Kasper; Boring, Sebastian

Published in:
CHI '18 Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems

DOI:
10.1145/3173574.3174138

Publication date:
2018

Document version
Publisher's PDF, also known as Version of record

It's a Wrap: Mapping On-Skin Input to Off-Skin Displays

Joanna Bergstrom-Lehtovirta, Kasper Hornbæk, Sebastian Boring
University of Copenhagen, Denmark
{joanna, kash, sebastian.boring}@di.ku.dk

ABSTRACT
Advances in sensing technologies allow for using the forearm as a touch surface to give input to off-skin displays. However, it is unclear how users perceive the mapping between an on-skin input area and an off-skin display area. We empirically describe such mappings to improve on-skin interaction. We collected discrete and continuous touch data in a study where participants mapped display content from an AR headset, a smartwatch, and a desktop display to their forearm. We model those mappings and estimate input accuracy from the spreads of touch data. Subsequently, we show how to use the models for designing touch surfaces to the forearm for a given display area, input type, and touch resolution.

Author Keywords
Skin input; touch surface design

ACM Classification Keywords
H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION
With advances in on-skin sensing technologies, the forearm has become a prominent surface for providing touch input to off-skin displays, such as smart watches [12, 21], mobile phones [10], smart glasses [18], and remote displays [2, 11]. To perform input on off-skin displays, there needs to be a mapping between the forearm's surface and the display's coordinate system (i.e., which location on the forearm corresponds to which location on the display). These mappings are commonly defined in an ad-hoc fashion. It is, however, unclear whether users perceive that mapping in the same way as the system (Figure 1).

Designing a mapping is a complex task, because the spatial relation between the forearm and a display differs in many ways: a display is often planar and rectangular while the forearm is irregular and deformable. For example, moving an on-screen cursor on a straight line to the left requires the knowledge of a straight line and the direction left on the skin, and a 10 cm distance on a display can be perceived differently around the wrist than on the thicker part of the arm near the elbow. To simplify the design task, previous work has treated the skin either as a planar surface with a straightforward mapping of touch points [7, 21], or used non-visual [1, 5, 6] or one-dimensional content [10, 11] on local areas on either the anterior [7, 8, 21] or posterior [10, 21] sides of the forearm. Such simplifications, however, can hamper interaction if they contradict the user's perception of the mapping between touches on the skin and the content on a display. In this paper, we contribute empirical findings on user-defined mappings (i.e., how do users *perceive* such mappings) and models describing those (i.e., how can input be *mapped* accordingly).

Based on common use of forearm input in previous research (e.g., [9, 11, 14, 18, 21]), we collected touch data in a study, where participants map off-skin displays (and their content) to their forearm. We chose three common display types for skin input, namely a smart watch, a desktop display, and an AR headset. In addition, we chose two different layouts (horizontal and vertical) and collected two input types (discrete and continuous touches). With the collected data, we:

1. Model the mappings between display (and content respectively) and touches on the forearm,
2. Evaluate the accuracy of our mappings across users, and
3. Describe how these mappings may change based on display and input types.

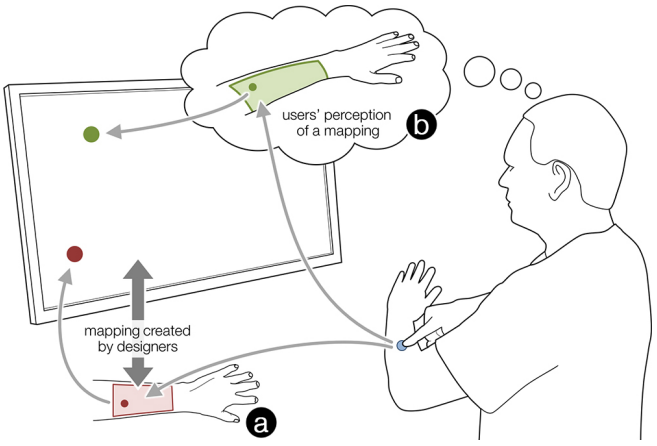

Figure 1. (a) The system's interpretation for mapping on-skin input to off-skin display differs from (b) user's perception of it. Such inconsistent mapping between the touch location on the arm and a content point on a display hampers interaction.

CHI 2018, April 21–26, 2018, Montreal, QC, Canada

ACM 978-1-4503-5620-6/18/04...$15.00.
https://doi.org/10.1145/3173574.3174138

RELATED WORK

The general problem we examine is how to map skin input to output. Relevant work has been carried out in sensing skin input as well as in designing touch target layouts for the skin.

Using the Forearm as an Input Surface

Previous work has presented a variety of sensing systems for skin-based interfaces that allow using the forearm as a touch surface. These include electronic tattoos [20], electrical sensing [21], acoustic sensing [8, 11], and optical sensing [7, 15]. The interfaces without visual content do not require a mapping between the touch locations and the content locations. The interfaces that allow direct touch input inherently use a 1-to-1 mapping.

Most of the sensing technologies, however, also allow and have been used for giving indirect input on the skin to off-skin displays [2, 3, 10, 18, 21]. *PalmRC* [2], for example, allows for using the palm as input surface for a distant television, and *PalmType* [18] as a keyboard for typing on smart glasses. Also the larger surface area of the forearm has been used to give input, for instance, to smart watches [9, 13, 21]. These systems require mappings between sensed on-skin touch locations and off-skin display content. However, it is unclear if the used mappings are effective for interaction.

An exception in this body of work is a system of Gannon et al. [3, 4] which processes touch around the entire forearm for sketching physical on-body 3D designs. The mapping they applied was effective, but on purpose straightforward: The output of touches on a 3D arm model needed to follow actual touches. Thus, a mapping between the irregular shape of the arm and a planar display surface was not required.

Although interfaces that use an ad-hoc mapping make simplifications that are sufficient for the applications they were intended for [3], such mappings may not be feasible for other purposes or at other locations on the forearm. For example, touch accuracy varies significantly across the forearm [21], and is lower than on external surfaces [7]. Although an actual sensor might perform with excellent precision, inaccuracies in input might still occur due to the simplifications the system makes regarding the irregular shape of the arm.

Designing Mappings for Off-Skin Displays

In addition to the ad-hoc mappings designed for a particular application or a study, previous work has investigated user-defined mappings. For example, how users design mappings based on skin-specific landmarks [1], how they would align the hand to match the directions of a display [2], or how they would map a keyboard layout to the shape of the palm [5, 18]. These studies cover one dimensional content, for instance, placing six to nine touch targets on the arm [10], local areas of skin on the palm [2, 18], location-independent gestures [19, 13, 17], and non-visual content [1, 5, 6]. However, mappings for indirect touch input on the forearm for visual content on off-skin displays have not yet been established. The purpose of our work is to suggest such mappings, and to provide insights on how common display and input types may influence those.

Mapping Touch on the Forearm

Placing touch targets inconsistently with users' perception and preferences can cause errors in tracking input and may result in confusion when displaying the corresponding output. While mapping rules that are in conflict with expectations may likely take longer time for users to learn, interaction can also benefit from matching those mappings, for instance, by guiding input [1, 6, 10] and enhancing finger and arm movements in input [16]. The possibilities for improving touch interaction on the arm suggest that both designers and sensor developers can benefit from models that quantify how users perceive interfaces on their skin.

DATA COLLECTION

The goal of collecting touch data was to find user's perceived mappings between display content and touch locations on the forearm. To address this goal, we set three requirements, which informed the design of our study. First, no cursor could be used in the experiment task for indicating a finger position on the display. A cursor would have necessitated a transfer function of our choice, thus giving a mapping instead of finding participant's perception of one. Second, touch samples were needed across the entire forearm's surface in order to model the mappings beyond smaller, local areas on the arm. To collect these, participants were instructed to make use of the entire forearm's surface in two experiment phases. This approach allows modeling the mappings on the entire forearm, but also generalizing on smaller areas by interpolation. Third, to gain insights from more realistic conditions, we collected mapping data with three common display types, and with both discrete and continuous touch. Furthermore, mapping data was collected using the three displays without the spatial constraints introduced by the previous requirement. While using the entire arm informs how content is mapped within its surface, giving up that restriction provides us insights about participant's choice of the alignment and size of the mapped display surface on the forearm.

Participants

Data were collected from 18 right-handed participants (4 females, with an average age of 26.8 years). All of them used a touchscreen phone regularly, eight used a tablet device regularly, and one used a smart watch. Six participants had tried AR or smart glasses prior to the experiment. The sizes of the participants' arms were measured as those influence how the touch data is processed. The circumference of the left forearm was on average 257.72 mm (SD = 20.94 mm), the width of the wrist 51.89 mm (SD = 4.21 mm), and the height 43.39 mm (SD = 4.29 mm).

Participants were instructed to wear a shirt that left the skin bare below the elbow. We also asked them to remove bracelets or watches prior to their participation. The study lasted approximately an hour, and participants received gifts as compensation for their time.

Design

The study followed a within-subjects design and consisted of three phases. Each participant performed those phases in the

Figure 2. The study setup: a) The layout of the cameras of an OptiTrack motion capture system to track the arm posture based on two green marker bodies attached to the wrist and to the elbow, and the fingertip location based on a yellow marker body attached to the nail. b) Visualizing touches on an arm model used in processing motion capture data. c) Conditions where the participant maps content from a desktop display, d) from a watch, and e) from an AR headset to the arm. The dot highlighted red is the point to be mapped in the current trial.

same order. Different conditions within phases were counterbalanced. The order of trials within each phase was randomized.

In the **device phase**, participants mapped display areas from three different *devices* to their forearm. Those devices were a desktop display, a smart watch, and an AR headset. We chose those devices for two reasons. First, they are commonly used in research on skin-based interfaces. Second, they represent different spatial features: a desktop display is stationary within the room, a watch is attached to the forearm, and the AR visualization follows the user's head movements. The intent of this phase was to allow for analyzing the effect of display type on location and size of the mapped areas. The order of devices within this phase was counterbalanced across participants. Because we asked participants in the latter phases to make use of the entire forearm, we chose this phase to be first, so to not influence their preferences with respect to touch surface size and location.

In the **main phase**, participants mapped content shown on a regular screen to their *entire forearm*. That content was shown in two orientations – horizontally and vertically. We chose these two distinct orientations to make use of two common postures in forearm input: the user has a primary view to the posterior side when forearm is held horizontally, and to the anterior side when it is held vertically. Both sides, however, were used in each condition, to allow us to observe the effects of both the postures and their rotation on mappings on the entire forearm. All participants used both orientations, and we counterbalanced that order across participants. This phase allowed us to model mappings between the entire forearm and display content, as well as to analyze input resolution with discrete touches.

In the **continuous touch phase**, participants mapped lines between two discrete content locations by giving *continuous* input (i.e., an on-skin stroke). We chose to also include continuous touch input, as it represents a common touch input style on the forearm besides discrete touches used in the two other phases. The collected data here allow for analyzing the effects of this common input type regarding both mapping and touch resolution. We chose to perform this phase last, so to not influence the mapping in the main phase, in which a similar display area was used.

Task

Participants were asked to map content from off-skin display surfaces to touch points on the forearm. In each phase, all targets were shown at the same time. During the first two phases, the point to be touched next was highlighted in red. After a touch was registered, the next target was highlighted after a two-second threshold (to avoid accidental activation). During the third phase (continuous touch), the start and end target were also highlighted in red and numbered with 1 (*start*), and 2 (*end*). The next two targets were highlighted after the same two-second threshold.

Procedure

Table 1 shows the main steps of data collection. At the beginning of the study, participants were equipped with three rigid marker bodies needed for motion capture (see Figure 3a–c). Then, before stepping into the three main phases, participants were allowed to try out using the skin as touch surface. During this training task, participants tapped 30 times on their forearm and were able to see the outcome illustrated as dots on a live model of the arm (see Figure 2b). This phase ensured that participants could see that even a light tap on the skin is registered as touch, and that these touches are accurately placed on our forearm model. Here, we describe the procedure of the three main phases in more detail.

First Phase: Devices

In the first phase the participants mapped areas from displays of the three devices (a desktop display, a watch, and an AR headset) to their arm. The devices were used one at a time to create a mapping for each.

The target to be mapped were displayed on a square area in a 4×4 grid (see Figure 2c–e). Participants could freely choose to map content (and the active target respectively) anywhere and in any scale and alignment onto their forearm. We only instructed them to maintain the order/layout of the displayed points during mapping.

Each target was repeated three times for each device. While the targets were randomized within each block, the first block started out by mapping the four corner points (in randomized order), so that participants implicitly set the boundary of their intended touch area. This ensured that participants thought about the mapping of the entire layout first. In the remaining two blocks, however, all targets were randomized. In total, we collected $3 \times 16 = 48$ targets per device per participant.

We chose a square area with a regular grid so that participants would only be minimally influenced in their decision with respect to rotation of the touch surface. A rectangular (or irregular) shape may have influenced participants so that they

would map those to the shape of the forearm. Furthermore, having the same layout on all used displays allowed us to compare the different devices rather than different layouts.

Main Phase: Entire Forearm

In the second phase the participants mapped content shown in both horizontal and vertical orientation on the desktop display onto their forearm. The two orientations were mapped one at a time. To indicate the different orientations, we chose a rectangular area with a 6×4 grid which contained all targets. Each target was mapped three times in random order, giving a total of $3 \times 6 \times 4 = 72$ touch samples per orientation per participant. Note that, unlike in the first phase, we did also fully randomize the first block.

In the horizontal condition, participants were instructed to map the left edge of the displayed area to the elbow and the right edge to the wrist. They were further instructed to make use of the *entire* forearm's surface. This is, the upper and lower edges should meet at the ulna (see the coordinate system and Figure 3d–e in the subsequent section), and the downward direction should correspond to the direction leaving the ulna to radius across the posterior side (and radius to ulna across the anterior side).

In the vertical condition, participants were instructed to map the lower edge to the elbow and the upper edge to the wrist. Thus, the left and right edges of the displayed area should meet at the ulna, and the direction from left to right should correspond to the direction from ulna to radius across the posterior side (and radius to ulna across the anterior side). In essence, the vertical layout was a 90° rotation of the layout.

We instructed participants by pointing out the reference edges on their forearm. The purpose of instructing those wrappings of the displayed area was to allow for sampling touches and modeling mappings beyond local areas (e.g., posterior side only) as well as across the entire forearm.

Third Phase: Continuous Touch

In the third phase the participants mapped pairs of targets (and the line between those respectively) on a desktop display by performing a continuous stroke from one target to another. In this phase, we used the horizontal layout from the second phase, and chose the outer targets (i.e., the targets on the edge) as start and end points. We instructed participants in the same way as in the second phase with respect to how the layout should be wrapped around the forearm.

In this phase, two targets were highlighted at once and numbered 1 and 2 to indicate the direction of the stroke. The trials covered all horizontal (4 rows) and vertical (6 columns) lines, as well as the two diagonals between opposite corners. Additionally, each pair of targets contained two directions. The order of target pairs (including their direction) was randomized and repeated three times. In total, we collected $3 \times 2 \times (6 + 4 + 2) = 72$ traces per participant.

Table 1. Steps of data collection, tasks, and purposes.

Step	Task	Purpose
Training Task	The participant touches the arm 30 times. The display shows that the touches are registered with dots on an arm model following user's movements real-time (Figure 2b).	Visualizing the arm and touches on it help the participant to understand the purpose of mapping.
1st Phase: Devices	The participant maps 16 points on a display to the forearm on the area of their choice with discrete touches and three repetitions (Figure 2c-e).	Allows analysis of the effects of common display devices on location and size of the mapped areas.
Main Phase: Entire Forearm	The participant maps 24 points oriented vertically or horizontally on a desktop display to the entire forearm with discrete touches and three repetitions.	Allows modeling the mappings across the entire forearm, as well as analysis of input resolution with discrete touches.
3rd Phase: Continuous touch	The participant maps pairs of points on a desktop display by tracing from an assigned point to another.	Allows analysis of the effects of continuous touch input on both mapping and touch resolution.

APPARATUS FOR COLLECTING TOUCH DATA

To allow for collecting accurate touch data and model the mappings in relation to the forearm's surface, we needed to track touch locations precisely on the forearm and describe those touches in a coordinate system relative to the forearm. Touch tracking on the arm has previously been done using depth cameras (e.g., [7, 15]) with varying accuracy and touch tolerance. As we were interested in the precise touch location, we opted for tracking both the arm and touching finger using an *OptiTrack* motion capture system. Such tracking systems, however, track individual retro-reflective markers (or a rigid marker body) instead of a full 3D model. For this reason, we had to approximate the forearm's shape using a minimally intrusive set of markers attached to the forearm. In this section, we first introduce the approximate model, followed by a description of how we use that model to precisely detect touch on the forearm. Finally, we outline the actual implementation of our apparatus.

Approximating the Forearm's Anatomy

In order to keep the forearm nearly marker-free (instead of giving both visual and tactile guidance to users through a large number of markers), we chose to attach only two rigid marker bodies to the forearm (see Figure 3a–b). We identified such locations for these that are outside the input area to minimally influence the mapping tasks, and which are easy to locate to keep the placement of the marker bodies and thus tracking of touch consistent across participants: the end of the ulna (below the elbow), and the wrist (see Figure 3d).

Shape: Between these two locations, we approximate the forearm's shape as a cone-like geometry, with a circular base (at the elbow) and an elliptical top (at the wrist). Note that we consider the circle at the base in the following as ellipse with equal major and minor radii: $a_{arm} = b_{arm}$. We first use the forearm's circumference and the wrist's width and height, alongside the markers' positions and orientations, to calculate the forearm's centerline. This line will also serve as x-axis, where 0.0 denotes the location of the elbow, and 1.0 that of the wrist. The shape of the forearm at any given location x_μ

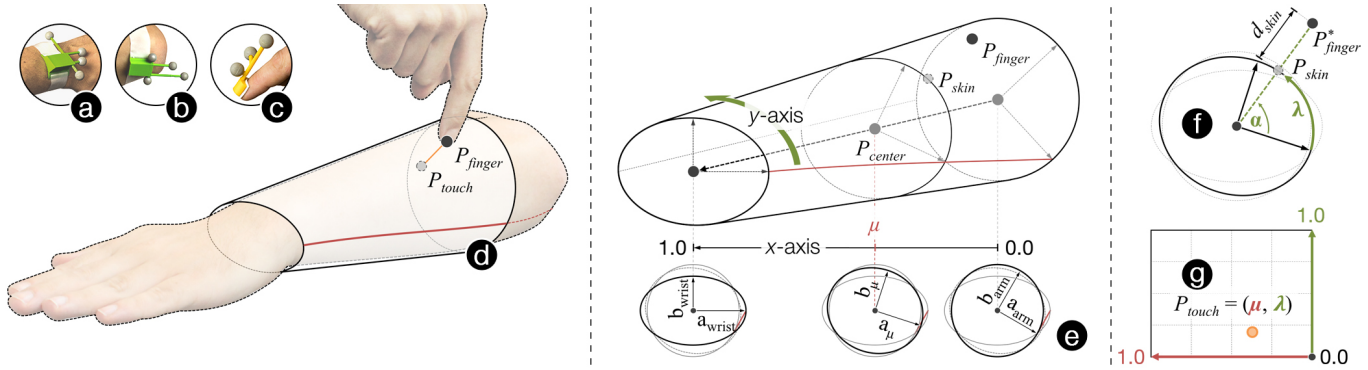

Figure 3. Rigid bodies with markers attached to the wrist (a), elbow (b), and fingernail (c) enable tracking of both the arm and fingertip location and orientation in 3D. The arm-centric coordinate system (d) on an arm allows precise touch tracking. On the middle: The x-axis of the coordinate system follows the ulna (in red), and the y-axis travels around the arm surface. The arm model (e) is interpolated along the x-axis from a circle in the base at the elbow to an ellipse at the wrist. Touch coordinates are mapped using the distance of the fingertip to the centerline of the arm model, and by projecting its position on local ellipse coordinates (f). This allows expressing the touch coordinates on a normalized 2D surface (g).

(where $x_\mu \in [0;1]$) is then an ellipse E_μ (with major radius a_μ and minor radius b_μ) using the following linear interpolation (see Figure 3e):

$$a_\mu = (1.0 - x_\mu) \cdot a_{arm} + x_\mu \cdot a_{wrist}$$
$$b_\mu = (1.0 - x_\mu) \cdot b_{arm} + x_\mu \cdot b_{wrist}$$

The rotation of each ellipse E_μ (relative to the elbow's initial rotation) is then also linearly interpolated. As both marker rotations are described as quaternions (through our tracking system), we interpolate the rotation of E_μ using *Slerp* (**s**pherical **l**inear int**erp**olation). Given the rotations q_{arm} and q_{wrist}, the resulting rotation q_μ of E_μ at x_μ is thus calculated as $q_\mu = slerp(q_{arm}, q_{wrist}, x_\mu)$, where $x_\mu \in [0;1]$.

Coordinate System: As described before, the *x*-axis is defined as the centerline. The *y*-axis, however, is of cylindrical nature. Here, we define the zero line as the line formed by the ulna. That is, any point located on the ulna has the coordinates $P(x, 0)$. As shown in Figure 3e, going *around* the arm then changes that *y*-coordinate, where it increases when rotating around the forearm's axis in clock-wise direction until it reaches the ulna again (where it will drop from 1.0 to 0.0).

Touch Detection

To detect touch, we attached another rigid body to the fingertip of the index finger (see Figure 3c). The origin of the rigid body's coordinate system is shifted, so that it is located at the actual fingertip. In the following, we will use the notation P_{finger} as the 3D position of the fingertip in real-world space, and P_{touch} as the 2D position on the forearm's surface.

For each frame coming from our tracking system, we first determine the closest point P_{center} to P_{finger} on the forearm's centerline. We then determine whether P_{center} is between the elbow's and wrist's center, and thus on the line segment between those two points. If P_{center} is not on that segment, or the distance to the centerline is too large (e.g., larger than twice the forearm's radius at the elbow), the finger is not touching the forearm's surface. Otherwise, we continue by normalizing P_{center} to $\mu \in [0;1]$. The parameter μ thus gives us the *x*-coordinate of P_{touch}.

Using the above-mentioned approximation, we can now use μ to calculate the ellipse E_μ. We further use the global transformation of E_μ (its position and orientation) to transform the fingertip into the ellipse's 2D coordinate system (now called P^*_{finger}) for subsequent calculations. In the ellipse's local coordinate system, we then calculate the angle α between the vector from the ellipse's origin to P^*_{finger} and the major axis. Using α, we calculate the projected position P_{skin} in the ellipse's local coordinate system (see Figure 3f):

$$P_{skin} = (a_\mu \cdot \cos\alpha, b_\mu \cdot \sin\alpha)$$

The distance d_{skin} between P_{skin} and P^*_{finger} can now be used for determining whether this is a touch event (distance smaller than a predefined threshold). If that is the case, the *y*-coordinate of the on-skin location P_{touch} is calculated as ratio λ between the arc length of P_{skin} and the ellipse's full perimeter (i.e., $\lambda \in [0;1]$), leading to a 2D on-skin location $P_{touch} = (\mu, \lambda)$. Figure 3g shows P_{touch} in the forearm's coordinate system, which can now be used for interaction.

Implementation

The central component of our apparatus is an OptiTrack motion capturing system, consisting of eight Flex13 cameras (56° field of view, 1280 × 1024 pixels, 120 fps). These cameras were connected (through two OptiHubs) to a desktop computer (2 XEON CPUs with 2.5 GHz each, 64 GB RAM, and 2 GeForce GTX 980 graphics adapters) running Windows 8.1. The software used during the study was written in C# (.NET 4.6), and interfaced with OptiTrack through NaturalPoint's NatNet SDK. Throughout the study, the OptiTrack system was running at 120 fps. We tracked the finger, elbow, and wrist through custom, 3D-printed marker bodies, which were attached using adhesive tape (see Figure 3a–c). Prior to the experiments, the OptiTrack system was calibrated to a global minimum accuracy of 0.1mm. If the system reported that tracking was no longer performing reliably within this accuracy, we calibrated the system again.

The desktop computer also drove the main display used during all phases of the study. The other two displays in the first phase were an Apple Watch (42 mm model, 312 × 390 px,

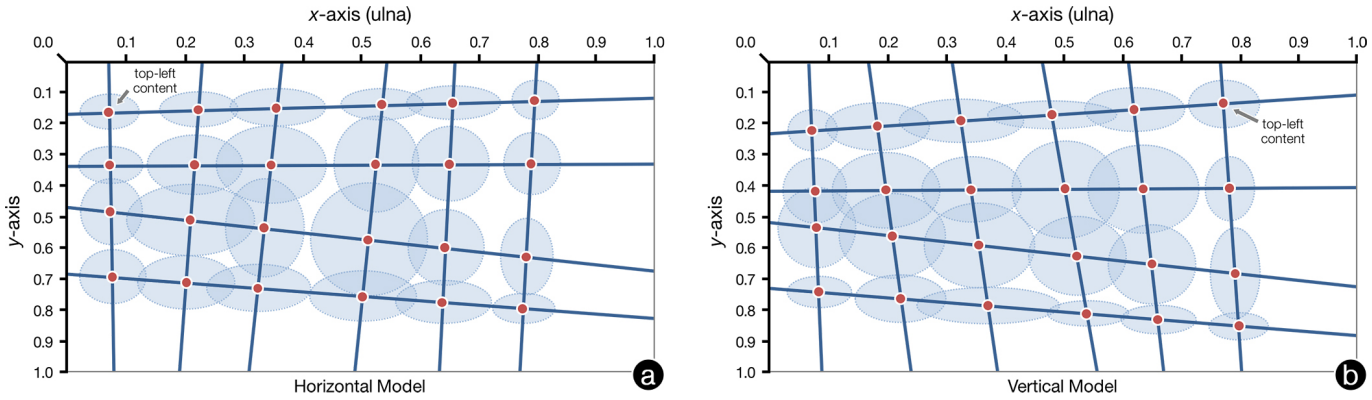

Figure 4. The linear models in blue lines describe the mappings participants created in the second phase of the study for the horizontally and vertically oriented content. The red dots are the intersections of these models, describing the average coordinates of the 24 mapped points on the forearm. The coordinate system and data units are described in the previous section.

watchOS 3), and a Microsoft HoloLens. The Apple Watch interfaced with an iPhone 6 over Bluetooth (both applications written in Objective-C). The HoloLens application was created in Unity. Both the iPhone 6 and the HoloLens were connected through WiFi (over TCP) to the main computer.

The tracked data was written in text files (separate for each phase), where we logged the raw positions and orientations of each rigid body, as well as the calculated touch points. In the first two phases, we only logged the moment of the touch. In the third phase, we logged all touch locations occurring during a continuous touch on the forearm.

To allow for accurate touch tracking, the software has to know the user's arm measures: prior to the study, the experimenter measured the arm's circumference to determine a_{arm} and b_{arm} (which are of equal length). The experimenter also measured the wrist's width (to determine a_{wrist}) and height (for b_{wrist}). Because our approach is an approximation of the forearm's shape, we further set (after experimentation) the thresholds to 10 mm for touch (i.e., a finger closer to the skin than this threshold results in touching the skin).

Quality of Arm Approximation and Touch Detection

The above-mentioned threshold of 10mm compensates for the forearm's surface being thicker (note, that there is no minimal negative threshold). The recorded touch point during the experiment is calculated once a finger leaves the touch threshold again. To further avoid triggering of repeated touch events or detecting a false touch event (i.e., where a user moves briefly inside the threshold but leaves again), we require the finger to be in a touch/no-touch state for at least 3 frames (25 ms at 120 fps). No false positive touches were detected during the experiment, indicating that the surface remained below the 10mm threshold over the approximation. This suggest at least a 10mm accuracy for the approximation.

Because we approximate the shape of the forearm, detected touch locations would slightly vary from the actual location on the forearm if we were to take the calculated position on the virtual, cylindrical approximation. In reality, some parts of the forearm's surface may slightly be inside the approximation, while others are outside. As shown in Figure 3f, d_{skin} (the distance between the actual finger location and the one calculated on the approximated surface) would thus be negative when the forearm's actual surface is thinner than our approximation (and positive when the forearm's surface is thicker, respectively). For this reason, we chose to use the closest point to the skin's surface as actual touch point (i.e., the point where P^*_{finger} is closest to P_{center} when moving below the threshold). Thus, our approximation has a negligible influence on accuracy.

MAPPINGS OF TOUCH SURFACES

In this section we formalize, analyze, and describe the mappings based on the collected touch data. To do so, we first model the mappings between the displayed content and touches on the forearm. Then, we use data from repeated trials to investigate how consistently these mappings hold – this informs us about the touch resolution that can be achieved on the forearm. Finally, we describe the effects of display devices on the size, location, and orientation of the mappings on the forearm, as well as the effects of continuous touches on mapping and touch accuracy.

Models of Mappings across the Forearm

We used touch data from the main phase to model the mappings on the entire forearm. We first started by removing outliers representing unintended taps. To do this, we fitted linear models on the grids of points for each participant and each condition across the three repetitions. During data collection, participants occasionally mentioned, for instance, that they had touched the "wrong" point (i.e., they would have mapped this point to another target on the display, but acted too quickly), which they noticed after touching the forearm. The linear models were used to find such unintended taps in mapping.

After fitting the models to the raw data, all touch points further than $1/4$ from the row models and further than $1/6$ from the column models were flagged as outliers, and the union of these row and column sets were then removed. Out of a total of 2592 samples collected in the second phase, 94 samples (or 3.6%) were removed as outliers.

Table 2. Coefficients for the linear models describing the mappings of horizontally and vertically oriented content.

	Horizontal models		Vertical models	
	a	b	a	b
Coefficients for $y = ax + b$				
Row 1	-0.0516	0.1683	-0.1234	0.2326
Row 2	-0.0083	0.3366	-0.0104	0.4184
Row 3	0.2039	0.4701	0.2085	0.5194
Row 4	0.1454	0.6832	0.1537	0.7310
Coefficients for $x = ay + b$				
Column 1	0.0095	0.0710	0.0228	0.0668
Column 2	-0.0390	0.2302	0.0710	0.1682
Column 3	-0.0564	0.3661	0.0761	0.3104
Column 4	-0.0543	0.5439	0.0943	0.4634
Column 5	-0.0271	0.6604	0.0598	0.6100
Column 6	-0.0299	0.7997	0.0355	0.7668

Models

We separately fitted linear models to the rows ($y = ax + b$) and columns ($x = ay + b$) of touch data mapped to the 6×4 grid of targets displayed to the participants. Linear models were chosen, because other models did not significantly increase the fit (and to avoid the risk of overfitting). We first fitted those models for each participant individually using a least squares method. The average linear models for both orientations across all participants are plotted in Figure 4 using the forearm's coordinate system described in the previous section. The coefficients for each of the 10 model lines and for both content orientations are reported in Table 2.

Model Fitness

The root mean squared errors (RMSE) describe the estimated errors of the models at the units of the coordinate system (between 0.0 and 1.0). For the linear models of the horizontal condition the mean RMSE is 0.078 and for the vertical condition the mean RMSE is 0.090.

Touch Accuracy

We use quantiles to estimate the feasible sizes of grid cells for accurate touch using the mappings. We first calculated the standard deviations for touches mapped for each target in the horizontal and vertical conditions in the main phase. From the standard deviations, we calculated the quantiles. A touch is estimated to lie outside the intervals described by quantiles in 5% of the cases. The quantiles are plotted as error ellipses in Figure 4 for both orientations, and the sizes of quantiles along the x- and y-axis for all 24 targets in both horizontal and vertical conditions are reported in Table 3.

The quantiles are smaller (that is, touch accuracy is higher) near the wrist and near the elbow compared to the space between those. Also, the posterior side yielded smaller quantiles than the anterior side (see Figure 4). By calculating the mean values of these quantiles for both axes separately, we estimated the number of touch points that could be accommodated on each row and column of a grid. For horizontal layouts, such estimation yielded a maximum of 7 touch points across the posterior (rows 1 and 2 in Figure 4a), 6 points on the anterior side of the forearm (rows 3 and 4 in Figure 4a), and 6, 5, 4, 4, 5, and 5 points on the six columns (from elbow to wrist). For vertical layouts, all rows could fit 6 targets (see Figure 4b), and the columns could fit 5, 4, 5, 5, 5 and 5 targets.

Although the estimation is conservative in rounding the number of the points down, it is still optimistic by both treating the unsampled areas with mean values, and by assuming the existing errors would not increase regardless of adding new touch points (which likely could happen).

Effect of Content Orientation on Mappings

The touch data show differences between the horizontally and vertically oriented content in the mappings as well as in touch accuracy. The row models describing the mapped targets along the forearm's length are shifted nearly 10% lower (i.e., further around) on posterior side of the forearm in the vertical orientation compared to the horizontal one. Furthermore, for the horizontal orientation the targets on the posterior side of the forearm (rows 1 and 2 in Figure 4a) are mapped nearly parallel to x-axis (the ulna), whereas they are more sloped for the vertical orientation. Figure 4 shows this effect with line ends to the left (the elbow) being closer for the vertical orientation (b) than for the horizontal one (a).

Closer lines (particularly for the vertical orientation towards the elbow) further result in higher probabilities of intersecting quantiles, thus having a negative effect on touch accuracy. The quantile intersections in Figure 4 shows this more clearly: better touch resolution can be achieved with the horizontal mapping, whereas the vertical one suffers from poor accuracy for targets near the elbow.

One surprising finding contained in our collected touch data was that a large space remains unused – particularly near the elbow (Figure 4), despite instructing participants to wrap the displayed area around the *entire* forearm. Interestingly, targets closer to the wrist were more equally distributed around the forearm.

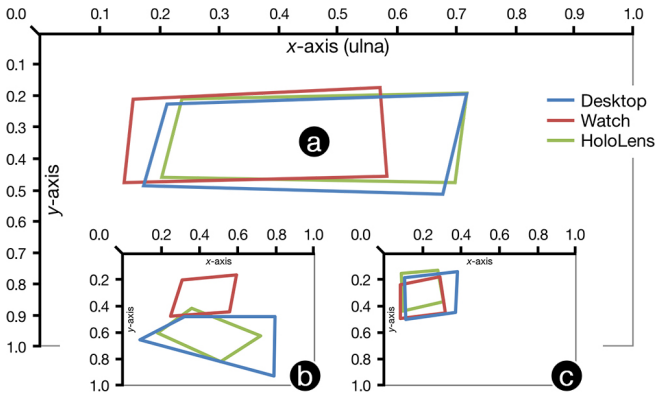

Figure 5. (a) Average boundaries for areas mapped from the three display devices, and (b-c) two examples of individual differences between the mappings of participants. (b) The participant has used the posterior side only for mapping an area from the watch, while using anterior side for the desktop display and AR headset. (c) The participant has mapped areas from all display devices to surfaces notably smaller than the average boundaries suggest.

Table 3. The quantile ranges for the each of the 24 mapped points in horizontal and vertical conditions. Touch points are estimated to lie outside these intervals described by quantiles in 5% of the cases. The rows and columns correspond to those depicted in Figure 4. The x -values are the ranges along x -axis (across the arm's length), and the y -values along the y -axis (around the arm).

Quantiles for horizontal orientation

	Column 1		Column 2		Column 3		Column 4		Column 5		Column 6	
Row 1	0.1005	0.1125	0.1340	0.1120	0.1601	0.1186	0.1401	0.1008	0.1593	0.1077	0.0794	0.1286
Row 2	0.1088	0.1194	0.1619	0.1935	0.1875	0.2478	0.1436	0.3095	0.1303	0.2393	0.0966	0.1972
Row 3	0.1049	0.2114	0.2180	0.2312	0.1335	0.3151	0.1992	0.3607	0.1283	0.2426	0.0887	0.2475
Row 4	0.1132	0.1742	0.1658	0.1697	0.1770	0.1516	0.1822	0.1624	0.1615	0.1394	0.1064	0.0963
	x	y	x	y	x	y	x	y	x	y	x	y

Quantiles for vertical orientation

	Column 1		Column 2		Column 3		Column 4		Column 5		Column 6	
Row 1	0.0839	0.1312	0.1757	0.1536	0.2101	0.1362	0.2198	0.0902	0.1708	0.1394	0.1172	0.1503
Row 2	0.1082	0.2113	0.1831	0.2456	0.1702	0.2105	0.1805	0.3141	0.1903	0.2848	0.0850	0.2027
Row 3	0.1312	0.2601	0.1813	0.2733	0.1886	0.2721	0.1653	0.2543	0.1467	0.2583	0.0832	0.2975
Row 4	0.1100	0.1035	0.1561	0.1562	0.2456	0.1132	0.1388	0.0803	0.1280	0.0962	0.0975	0.0879
	x	y	x	y	x	y	x	y	x	y	x	y

Display Devices

To analyze the effects of different display types on mapped areas, we use the data from the device phase. We examined the location, size, and shape of those mappings. We first calculated the boundaries of the mapped surfaces from the average touch coordinates of the corner points for each participant individually. The differences reported in the following were tested using Bonferroni-corrected paired sample t-tests.

Location

Plots of the surface boundaries and centroids were used for comparing the locations of the mapped areas. Interestingly, the posterior side was used the most: only in 4 occasions out of the total of 54 the participants used the anterior side.

Although the average boundaries between displays were similar, they varied within participants (see Figure 5b–c). For example, two participants used the anterior side of the forearm for mapping the area displayed on the AR headset, and another one used that side for the desktop display. All but one participant mapped an area from the watch to a surface next to it (i.e., on the posterior side of the forearm). Finally, none of the mapped areas spanned across both sides of the forearm.

The location of the area mapped for the watch differed significantly ($p < 0.001$) from the two other displays: the average x-coordinate of the centroid for the watch was 0.36 (display: 0.45; AR headset: 0.47). This shift (pushing the area further toward the elbow) might be explained by the watch not being worn when other displays were used. The centroid's y-coordinate did not differ across displays.

Finally, the surfaces mapped for the AR headset had the most variation in location: the standard deviation of the centroids of those along the x-axis was 0.13, while it was only 0.08 for the watch.

Size & Shape

When looking at the size of the mapped areas, we found that the mapped area for the desktop display was significantly larger than that for the watch ($p = 0.016$). However, we found no other significant differences between the sizes of the mappings.

In the watch condition the height of the mapped surfaces remained the same across the arm, whereas in the desktop condition the surfaces were narrower close to the elbow than near the wrist. This implies that the participants scale the width in proportion of the arm width in the watch conditions, but apply more of a 1-to-1 mapping with the desktop.

Orientation

All surfaces mapped from the three devices to the posterior side were oriented so that the top left corner of the display was mapped closest to $P = (0, 0)$ on the forearm, and the top right corner closest to $P = (1, 0)$. This follows a mapping similar to the horizontal condition in the second phase. In contrast, the surfaces on the anterior side were also oriented similar to horizontal condition despite the need to then rotate the view when mapping.

Continuous Input

The data from the continuous touch phase was used to compare the mappings between this input type and discrete input used in the main phase. The touch traces collected in the continuous touch phase followed a similar distribution of points across the forearm's surface as for discrete touches (the horizontal condition). This suggests that the input style has a lesser effect on surface mappings than the visual features (i.e., content) of the displayed surface. Continuous touch input, furthermore, allowed participants to more consistently follow the created mappings than with discrete touches: The RMSE of the models for trace mappings was 0.049.

USING THE MODELS IN PRACTICE

Having developed our models, we now show how those can be used in practice. We first outline how a mapping is established given the coefficients of our models (see Table 2). We then show a visualization tool that supports the models' use in practice. Finally, we illustrate an example case comparing the effects of 1-to-1 mapping (without a model) and a mapping using the horizontal model for on-skin interaction.

Using the Coefficients for Mapping Touch Input

We use the coefficients and corresponding formulae (i.e., either $y = ax + b$ for rows; or $x = ay + b$ for columns) to

Figure 6. a) Scrolling the map to center New York by swiping horizontally on the arm. b) 1-to-1 mapping results to a downward drift on vertical direction on the display, and a transition too short on horizontal direction. c) using the models allows to map touch to the content on a display as users intended to.

calculate lines and intersections. The intersections are then being used to create the deformed grid G_{model} of the corresponding model. Additionally, we set up a template of the original items $G_{template}$: for the horizontal model, that grid has 4 rows and 6 columns, whereas the grid has 6 rows and 4 columns for the vertical model. The elements in $G_{template}$ are normalized from 0.0 to 1.0. For example, the grid location in the second row and fourth column has the normalized coordinates $e_{2,4} = (0.6, 0.33)$ in $G_{template}$. Finally, we create grid cells C for both the template grid and the model grid, where $C_{i,j} = [e_{i,j}, e_{i,j+1}, e_{i+1,j+1}, e_{i+1,j}]$.

For each grid cell, we now determine the homography $H_{i,j}$ (a transformation matrix) between that cell in the template grid and the same cell in the model grid. When a touch point P (or hover respectively) occurs, one first has to identify the grid cell in G_{model} that point is located in. Once that cell $C_{i,j}$ is found, the final touch location P' is calculated with $P' = H_{i,j} \cdot P$. The calculated point is now using the model, where both x- and y-coordinate are within 0.0 and 1.0. One can use these normalized coordinates as input to any screen of arbitrary width and height.

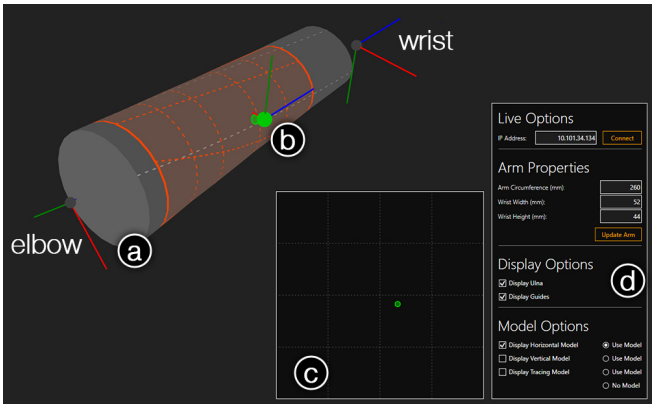

Figure 7. A screenshot of the visualizer contained in our toolkit: (a) the virtual representation of the approximation of the forearm including the horizontal model overlaid on the arm. (b) The finger's location. (c) The normalized and transformed location of the touching finger using the model. (d) The control panel, which allows for changing visualization options, and the arm's physical properties. One can also connect to a tracking system to switch to live view.

Visualizing the Models

The support the use of models in practice by designers, we developed a tool. Besides the calculations and correct application of our models, the tool contains a visualizer. Figure 7 shows the visualizer in action: the approximation of the arm and the selected model (here: horizontal) given the parameters arm's *circumference*, wrist *width* and *height* (a); the visualizer also allows for controlling a finger (b), which is is correctly transformed onto the normalized grid (c); users can change the arm's properties, the model to display, and which model to use for mapping (d).

Additionally, the toolkit supports using live tracking data. Currently, it supports an OptiTrack motion capturing system. The toolkit further allows subscribing to touch and hover events, which may be used in any application as input. Alongside this paper, the sources and binaries of the toolkit, the OptiTrack Motive project, as well as the OpenCAD files for 3D printing the rigid bodies for motion capture are available for download on our homepage [1].

An Application Example

We also built a simple application to demonstrate the effect of the models (see Figure 6) compared to a simple 1-to-1 mapping. In this application, a user interacts with a map on a larger display using on-skin input. The user wants to bring New York from the left side (a) to the center of the display (c). The user drags across the forearm in a way which he perceives as horizontal. However, the 1-to-1 mapping results in a different view (Figure 6b): (1) there is a drift in y-direction, causing New York to drop to end up further down on the screen; and (2), when approaching the center along the x-axis does not move New York as far as expected less than expected.

There are two reasons for this behavior: first, on the anterior side of the forearm, the lines have a slope of 0.2039 (see Figure 4 and Table 2 in the model-section), causing the drift downward. This means that a horizontal swipe from elbow

[1] https://github.com/DIKU-HCI/SkinTouch

to wrist causes an unwanted shift on of over 20% along the vertical direction on the display. Second, the perceived touch area does not extend to the whole length of the arm, and the points along *x*-axis are not equally distributed, resulting in less transition on the display along the *x*-axis. By applying the models to map the touch to the display content, the interface centers the target city on the map as planned with one horizontal trace across the arm (Figure 6c).

DISCUSSION

Mapping skin input to off-skin displays is a challenge for on-skin interaction. Interaction can be improved by employing mappings that more closely match a user's perception. By applying mappings that are consistent with a user's perception, the content on the display can be interacted with touches on the forearm in an unambiguous manner.

Our study revealed such mappings between on-skin input and off-skin displays. We collected touch data from participants mapping content from common display types to their arm, using both discrete and continuous touch input. Skin-based interface designs can benefit from the derived models.

Implications for Skin-based Interfaces

The main contribution of this paper are two models – one for horizontal and one for vertical content orientation – describing the participant-created mappings between input on the entire forearm and a display area (and content therein). These models show large differences between a user's perception and a straightforward mapping in both dimensions.

First, a 1-to-1 mapping assumes an evenly distributed grid of lines across the display. As an example, when distributing 4 rows evenly for horizontal layouts, those rows would be spaced 0.25 apart from each other. The coefficients b denote the intercept (with the *y*-axis) of each row and its corresponding line, thus showing how the model largely differs from a straightforward 1-to-1 mapping. Using the horizontal mapping, for example, the coefficient for the fourth row is $b = 0.6832$ (Table 2). In a 1-to-1 mapping this point would be $3 \times 0.25 + 0.125 = 0.875$. Thus, close to elbow the difference between the mappings is $0.875 - 0.6832 = 0.1918$, indicating a 19.2% mismatch between the mappings in relation to the display's height. Additionally, the user's perception of the location of this row is closer to the third than the fourth row in a 1-to-1 mapping (coefficient for the fourth row was $b = 0.6832$, and a 1-to-1 point 0.875, while the 1-to-1 point for the third row is 0.625). Therefore, such touch point would likely be falsely classified as the third one without the use of these models.

Second, a 1-to-1 mapping assumes no deviation across the grid lines. For example, a row of mapped points would stay at a constant height on a display. The row model coefficient a represents the vertical change in the mapping across the display from left to right. Using the third row, this coefficient is $a = 0.2039$ (Table 2), which corresponds to a 20.4% change of mapped points in the vertical dimension between the left and the right edge of the display. The mapping changes from the intercept point (for the third row, the Table 2 shows $b = 0.4701$) on the left edge to $ax + b = 0.2039 \times 1 + 0.4701 = 0.674$ on the right. Therefore, the mismatch changes as one moves towards the wrist, due to the slopes of each row.

We consider these effects to be of significance for real-world applications. There are deviations of over one fifth of the display's width and height, which can notably hamper interaction (the example application shows this effect as well). The models presented here contribute to compensating for the difference between a user's perception and a straightforward, 1-to-1 mapping of skin input to off-skin displays.

Limitations and Future Work

While these models address an important issue for using the forearm as a touch surface, we acknowledge that there are some limitations with respect to their generalizability and applicability to some of the open problems in forearm input.

First, the models describe the mappings in particular display conditions, namely with horizontally and vertically oriented content. Content may change its orientation during interaction, and so might the arm (to adapt for that change). While our models support both orientations, smooth transitions have not yet been established. Future work on how to adapt the mappings for changing types of displays and content could generalize the models for an even larger set of real-world applications.

Second, although we sampled touch data and arm postures in conditions requiring rotations, the models do not explain user behavior in interacting around the forearm. There are two reasons for this: (1) the mappings did not extend fully around the forearm, and (2) the tasks did not involve full rotations (i.e., users could perform tasks without rotating the forearm). Future work should thus also address rotations to open up possibilities for extending the input space and in guiding input, for instance, in bimanual interaction [16].

Third, the estimates for touch accuracy on these mappings do not yet allow optimization of touch target layouts on the forearm. To overcome this limitation, future work needs to address two challenges: more touch data is needed on a larger number of sample points around the *entire* forearm, and the changes in user performance on non-grid layout types as well as the cumulative effects of adding touch points on accuracy need to be investigated.

Finally, while this paper describes the basic effects of using continuous touch and three displays representing different spatial features, the mappings within those areas still call for further attention.

ACKNOWLEDGEMENTS

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement 648785). The work was supported by KAUTE Foundation and Ulla Tuominen Foundation.

REFERENCES

1. Bergstrom-Lehtovirta, J., Boring, S., and Hornbæk, K. Placing and recalling virtual items on the skin. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ACM (2017), 1497–1507.
2. Dezfuli, N., Khalilbeigi, M., Huber, J., Özkorkmaz, M., and Mühlhäuser, M. PalmRC: leveraging the palm surface as an imaginary eyes-free television remote control. *Behaviour & Information Technology 33*, 8 (2014), 829–843.
3. Gannon, M., Grossman, T., and Fitzmaurice, G. Tactum: a skin-centric approach to digital design and fabrication. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ACM (2015), 1779–1788.
4. Gannon, M., Grossman, T., and Fitzmaurice, G. ExoSkin: On-Body Fabrication. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ACM (2016), 5996–6007.
5. Gustafson, S., Holz, C., and Baudisch, P. Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, ACM (2011), 283–292.
6. Gustafson, S. G., Rabe, B., and Baudisch, P. M. Understanding palm-based imaginary interfaces: the role of visual and tactile cues when browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2013), 889–898.
7. Harrison, C., Benko, H., and Wilson, A. D. OmniTouch: wearable multitouch interaction everywhere. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, ACM (2011), 441–450.
8. Harrison, C., Tan, D., and Morris, D. Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI conference on human factors in computing systems*, ACM (2010), 453–462.
9. Laput, G., Xiao, R., Chen, X., Hudson, S. E., and Harrison, C. Skin buttons: cheap, small, low-powered and clickable fixed-icon laser projectors. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*, ACM (2014), 389–394.
10. Lin, S.-Y., Su, C.-H., Cheng, K.-Y., Liang, R.-H., Kuo, T.-H., and Chen, B.-Y. Pub-point upon body: exploring eyes-free interaction and methods on an arm. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, ACM (2011), 481–488.
11. Mujibiya, A., Cao, X., Tan, D. S., Morris, D., Patel, S. N., and Rekimoto, J. The sound of touch: on-body touch and gesture sensing based on transdermal ultrasound propagation. In *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces*, ACM (2013), 189–198.
12. Ogata, M., and Imai, M. SkinWatch: skin gesture interaction for smart watch. In *Proceedings of the 6th Augmented Human International Conference*, ACM (2015), 21–24.
13. Ogata, M., Sugiura, Y., Makino, Y., Inami, M., and Imai, M. SenSkin: adapting skin as a soft interface. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*, ACM (2013), 539–544.
14. Oh, U., and Findlater, L. A Performance Comparison of On-Hand versus On-Phone Nonvisual Input by Blind and Sighted Users. *ACM Transactions on Accessible Computing (TACCESS) 7*, 4 (2015), 14.
15. Sridhar, S., Markussen, A., Oulasvirta, A., Theobalt, C., and Boring, S. Watchsense: On-and above-skin input sensing through a wearable depth sensor. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ACM (2017), 3891–3902.
16. Strohmeier, P., Burstyn, J., and Vertegaal, R. Effects of display sizes on a scrolling task using a cylindrical smartwatch. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, ACM (2015), 846–853.
17. Tung, Y.-C., Hsu, C.-Y., Wang, H.-Y., Chyou, S., Lin, J.-W., Wu, P.-J., Valstar, A., and Chen, M. Y. User-defined game input for smart glasses in public space. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ACM (2015), 3327–3336.
18. Wang, C.-Y., Chu, W.-C., Chiu, P.-T., Hsiu, M.-C., Chiang, Y.-H., and Chen, M. Y. PalmType: Using palms as keyboards for smart glasses. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM (2015), 153–160.
19. Weigel, M., Mehta, V., and Steimle, J. More than touch: understanding how people use skin as an input surface for mobile computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM (2014), 179–188.
20. Weigel, M., Nittala, A. S., Olwal, A., and Steimle, J. Skinmarks: Enabling interactions on body landmarks using conformal skin electronics. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ACM (2017), 3095–3105.
21. Zhang, Y., Zhou, J., Laput, G., and Harrison, C. SkinTrack: Using the Body as an Electrical Waveguide for Continuous Finger Tracking on the Skin. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ACM (2016), 1491–1503.